Series PMC-LX/SX
Virtex-4 Based FPGA
PMC Module

USER'S MANUAL

ACROMAG INCORPORATED
30765 South Wixom Road
P.O. BOX 437
Wixom, MI 48393-7037 U.S.A.

Tel: (248) 295-0310
Fax: (248) 624-9234

Copyright 2010, Acromag, Inc., Printed in the USA.
Data and specifications are subject to change without notice.

8500-795-C10H013

TABLE OF CONTENTS

IMPORTANT SAFETY CONSIDERATIONS
You must consider the possible negative effects of power, wiring, component, sensor, or software failure in the design of any type of control or monitoring system. This is very important where property loss or human life is involved. It is important that you perform satisfactory overall system design and it is agreed between you and Acromag, that this is your responsibility.

The information of this manual may change without notice. Acromag makes no warranty of any kind with regard to this material, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. Further, Acromag assumes no responsibility for any errors that may appear in this manual and makes no commitment to update, or keep current, the information contained in this manual. No part of this manual may be copied or reproduced in any form without the prior written consent of Acromag, Inc.

1.0 General Information

KEY FEATURES…… 4
PCI INTERFACE FEATURES…… 5
Board DLL Control Software…… 6
Board VxWORKS Software…… 6
Board QNX Software…… 6

2.0 PREPARATION FOR USE

UNPACKING AND INSPECTION…… 7
CARD CAGE CONSIDERATIONS…… 7
BOARD CONFIGURATION…… 7
Default Hardware Configuration…… 7
Front Panel I/O…… 7
Non-Isolation Considerations…… 9

3.0 PROGRAMMING INFORMATION

GETTING STARTED…… 9
PCI CONFIGURATION ADDRESS SPACE…… 10
Configuration registers…… 11
MEMORY MAP…… 12
Flash Configuration…… 14
PCI bus to Xilinx Configuration…… 15
Configuration Status Register…… 15
Configuration Control Register…… 16
Configuration Data…… 16
Flash Status1 Register…… 17
Flash Status2 Register…… 17
Flash Read…… 17
Flash Reset…… 17
Flash Start Write…… 18
Flash Erase Sector…… 18
Flash Erase Chip…… 18
Flash Data Register…… 19
Flash Address Register…… 19
Reset Register…… 20
Rear Connector Read Register…… 21
Rear Connector Write Register…… 21
DMA Control Register…… 22
PLX PCI9656 DMA Setup Registers…… 22
DMA Transfer Size Registers…… 23
DUAL-PORT MEMORY …… 24
Dual-Port SRAM Control Register…… 24
Dual-Port SRAM Internal Address Register…. 25

Dual-Port SRAM DMA Threshold Registers…. 25
Dual-Port SRAM Address Reset Registers….. 26
Dual-Port SRAM Read Registers…….... 26
DDR SDRAM MEMORY 27
DDR SDRAM Control Register…….... 27
DDR SDRAM Address Register…….... 28
DDR SDRAM Read Registers…….... 30
DDR SDRAM Write Registers…….... 31
DDR SDRAM Mask Register…….... 32
STATIC RAM MEMORY…….... 32
Dual Port Memory…….... 32
Dual Port SRAM DMA Transfer Example…….. 33
PCI9656 REGISTERS…….... 34

4.0 THEORY OF OPERATION

PCI INTERFACE LOGIC…….. 35
NOT USED PCI9656 FUNCTIONS…….. 35
SYNCHRONOUS DUAL-PORT SRAM…….. 36
SERIAL EEPROM…….. 36
CLOCK CONTROL…….... 37

5.0 SERVICE AND REPAIR

SERVICE AND REPAIR ASSISTANCE…….... 37
PRELIMINARY SERVICE PROCEDURE…….. 37
WHERE TO GET HELP…….. 37

6.0 SPECIFICATIONS

PHYSICAL..... 38
ENVIRONMENTAL…….. 38
FPGA SPECIFICATIONS…….. 39
DUAL PORT SRAM…….. 39
PCI LOCAL BUS INTERFACE…….. 40

DRAWINGS

4502-054 BLOCK DIAGRAM..... 41
4502-053 J7 EXTERNAL POWER LOCATION …… 42

Trademarks are the property of their respective owners.

RELATED PUBLICATIONS

The following manuals and part specifications provide the necessary information for in depth understanding of the AX board.

Virtex-4 Data Book	**http://www.xilinx.com**
PCI 9656 Data Book	**http://www.plxtech.com**
IDT70T3519S133 Spec.	**http://www.idt.com**
MT46V32M16 Spec	**http://www.micron.com**
CY2305 Specification	**http://www.cypress.com**

1.0 GENERAL INFORMATION

The re-configurable PMC-LX modules use the Xilinx Virtex-4 LX FPGA. The re-configurable PMC-SX modules use the Xilinx Virtex-4 SX FPGA. Re-configuration of the FPGA is possible via a direct download into the Xilinx FPGA over the PCI bus. In addition, on board flash memory can be loaded with FPGA configuration data for automatic Xilinx configuration on power-up. Flash programming is also implemented over the PCI bus.

The example design includes an interface to the user rear I/O and front I/O connectors and an example memory interface controller to 32M x 32-bit DDR-SDRAM. Also, the example design includes an interface to the 256K x 36-bit SRAM with DMA hardware support.

Table 1.1: *The PMC-LX/SX boards are available in the standard temperature range*

MODEL	FPGA	OPERATING TEMPERATURE RANGE
PMC-LX40	XC4VLX40	0°C to +70°C or Conduction Cooled 0°C to +70°C
PMC-LX60	XC4VLX60	0°C to +70°C or Conduction Cooled 0°C to +70°C
PMC-SX35	XC4VSX35	0°C to +70°C or Conduction Cooled 0°C to +70°C

KEY FEATURES

- **Reconfigurable Xilinx FPGA** – In system configuration of the FPGA is implemented through a flash configuration device or via the PCI bus. This provides a means for implementation of custom user defined designs.
- **32M x 32 DDR-SDRAM –** A 32M x 32-bit double data rate (DDR) dynamic random-access memory (DRAM) is directly accessed through the Xilinx device. Read and write accesses to the DDR-SDRAM are burst oriented.
- **256K x 36 Dual-Port SRAM –** A 256K x 36-bit dual-port static random access memory (SRAM) is included. One port of the SRAM provides a direct link from the PCI bus to the SRAM memory. The second port of the SRAM provides a direct link to the Xilinx FPGA.
- **Interface to Front Multifunction Modules –** Various mezzanine modules, ordered separately, allow the user to select the Front I/O required for their application.
- **Interface to Rear P4 Connector –** Bank 5 of the FPGA is directly connected to 64 pins of the rear P4 connector. All 2.5volt IO standards supported by the Virtex 4 device are available. The example design provides low voltage differential signaling as 16 LVDS input and 16 LVDS output signals.
- **Write Disable Jumper –** User configurable flash and EEPROM board memory can be hardware write disabled by removal of an on board zero ohm surface mount resistor.
- **Example Design Provided –** The example VHDL design includes implementation of the PCI9656 Local bus interface, control of digital I/O,

SRAM read/write interface logic, and DDR-SDRAM memory interface controller.

PCI INTERFACE FEATURES

- **PCI Bus Master –** The PCI9656 PCI interface chip becomes the bus master to perform DMA transfers.
- **DMA Operation –** The PCI9656 supports two independent DMA channels capable of transferring data from the PCI to Local bus and Local to PCI bus. The example design implements DMA block and demand modes of operation.
- **32, 16, 8-bit I/O** - Register Read/Write is performed through data transfer cycles in the PCI memory space. All registers can be accessed via 32, 16, or 8-bit data transfers.
- **Compatibility** – Complies with PCI Local Bus Specification Revision 2.2. Provides one multifunction interrupt. Board is 5V or 3.3V signaling compliant. The voltage provided on the PCI connector VIO pins determines the operating voltage of the PCI bus.
- **Supply Voltage Requirement –** The board requires that 3.3 volts external power be provided on the 3.3 volt signal lines of the PCI bus connector.

ENGINEERING DESIGN KIT

Acromag provides an engineering design kit for the LX/SX boards (sold separately), a "must buy" for first time LX/SX module purchasers. The design kit (model PMC-LX/SX-EDK) provides the user with the basic information required to develop a custom FPGA program for download to the Xilinx FPGA. The design kit includes a CD containing: schematics, parts list, part location drawing, example VHDL source, and other utility files. The LX/SX modules are intended for users fluent in the use of Xilinx FPGA design tools.

BOARD DLL CONTROL SOFTWARE

Acromag provides a software product (sold separately) to facilitate the development of Windows (2000/XP/Vista/7®) applications accessing Acromag PMC and XMC I/O board products, PCI and PCIe I/O Cards, and CompactPCI I/O Cards. This software (Model PCISW-API-WIN) consists of low-level drivers and Windows 32 Dynamic Link Libraries (DLLs) that are compatible with a number of programming environments including Visual C++™, Visual Basic®, Borland C++ Builder® and others. The DLL functions provide a high-level interface to boards eliminating the need to perform low-level reads/writes of registers, and the writing of interrupt handlers.

BOARD VxWORKS SOFTWARE

Acromag provides a software product (sold separately) consisting of board VxWorks® software. This software (Model PMCSW-API-VXW) is composed of VxWorks® (real time operating system) libraries for all Acromag PMC and XMC I/O board products, PCI and PCIe I/O Cards, and CompactPCI I/O Cards. The software is implemented as a library of "C" functions which link with existing user code to make possible simple control of all Acromag PCI and PCIe boards.

BOARD QNX SOFTWARE

Acromag provides a software product (sold separately) consisting of board QNX® software. This software (Model PMCSW-API-QNX) is composed of QNX® (real time operating system) libraries for all Acromag PMC and XMC I/O board products, PCI and PCIe I/O cards, and CompactPCI I/O cards. The software supports X86 PCI bus only and is implemented as library of "C" functions which link with existing user code to make possible simple control of all Acromag PCI and PCIe boards.

BOARD Linux SOFTWARE

Acromag provides a software product consisting of board Linux® software. This software (Model PMCSW-API-LNX) is composed of Linux® libraries for all Acromag PMC and XMC I/O board products, PCI and PCIe I/O cards, and CompactPCI I/O cards. The software supports X86 PCI bus only and is implemented as library of "C" functions which link with existing user code to make possible simple control of all Acromag PCI and PCIe boards.

2.0 PREPARATION FOR USE

UNPACKING AND INSPECTION

Upon receipt of this product, inspect the shipping carton for evidence of mishandling during transit. If the shipping carton is badly damaged or water stained, request that the carrier's agent be present when the carton is opened. If the carrier's agent is absent when the carton is opened and the contents of the carton are damaged, keep the carton and packing material for the agent's inspection.

For repairs to a product damaged in shipment, refer to the Acromag Service Policy to obtain return instructions. It is suggested that salvageable shipping cartons and packing material be saved for future use in the event the product must be shipped.

This board is physically protected with packing material and electrically protected with an anti-static bag during shipment. However, it is recommended that the board be visually inspected for evidence of mishandling prior to applying power.

CAUTION
SENSITIVE ELECTRONIC DEVICES
DO NOT SHIP OR STORE NEAR STRONG ELECTROSTATIC, ELECTROMAGNETIC, MAGNETIC OR RADIOACTIVE FIELDS

WARNING: This board utilizes static sensitive components and should only be handled at a static-safe workstation.

CARD CAGE CONSIDERATIONS

Refer to the specifications for loading and power requirements. Be sure that the system power supplies are able to accommodate the power requirements of the system boards, plus the installed Acromag board, within the voltage tolerances specified.

In an air cooled assembly, adequate air circulation must be provided to prevent a temperature rise above the maximum operating temperature and to prolong the life of the electronics. If the installation is in an industrial environment and the board is exposed to environmental air, careful consideration should be given to air-filtering.

In a conduction cooled assembly, adequate thermal conduction must be provided to prevent a temperature rise above the maximum operating temperature.

IMPORTANT: *Adequate air circulation or conduction cooling must be provided to prevent a temperature rise above the maximum operating temperature.*

BOARD CONFIGURATION

Remove power from the system before installing board, cables, termination panels, and field wiring.

Default Hardware Configuration

The board may be configured differently, depending on the application. When the board is shipped from the factory, it is configured as follows:

- The default configuration of the programmable software control register bits at power-up are described in section 3.
- The control registers must be programmed to the desired configuration before starting data input or output operation.

Front Panel Field I/O Connector

The front panel connector provides the field I/O interface connections via optional mezzanine I/O modules, purchased separately.

Rear P4 Field I/O Connector

The rear I/O P4 PMC connector connects directly to bank 5 of the FPGA. The bank 5 VCCO pins are powered by 2.5 volts and thus will support the 2.5 volt IOStandards. The IOSTANDARD attribute can be set in the user constraints file (UCF) For example, rear I/O can be defined for LVCMOS25 (low voltage CMOS). The example design defines the rear I/O to LVDS_25 in the user constraints file. The 2.5 volt IOStandards available are listed in table 6-38 of the Virtex 4 User Guide.

The example design defines the rear I/O connector with 28 LVDS I/O pairs with an additional 4 LVDS input only pairs. The LVDS input only pairs are (RP_IO0, RN_IO1), (RP_IO10, RN_IO11), (RP_IO30, RN_IO31), and (RP_IO46, RN_IO47). The LVDS pairs are arranged in the same row in table 2.1. For example, RP IO0 and RN IO1 form a signal pair. The RP identifies the plus input while the RN identifies the negative input.

Table 2.1: *Board Rear Field I/O Pin Connections*

The example design implements 2.5volt LVDS I/O to the rear connector. Signal pairs are routed to pins (1,2), (3,4) etc.

Ch.	Pin Description	Pin	Pin Description	Pin
0	RP_IO0 (LVDS Input Only)	1	RN_IO1 (LVDS Input Only)	2
1	RP_IO2	3	RN_IO3	4
2	RP_IO4	5	RN_IO5	6
3	RP_IO6	7	RN_IO7_VREF	8
4	RP_IO8	9	RN_IO9	10
5	RP_IO10 (LVDS Input Only)	11	RN_IO11 (LVDS Input Only)	12
6	RP_IO12	13	RN_IO13	14
7	RP_IO14	15	RN_IO15	16
8	RP_IO16	17	RN_IO17	18
9	RP_IO18	19	RN_IO19	20
10	RP_IO20	21	RN_IO21	22
11	RP_IO22	23	RN_IO23_VREF	24
12	RP_IO24	25	RN_IO25	26
13	RP_IO26	27	RN_IO27	28
14	RP_IO28	29	RN_IO29	30
15	RP_IO30 (LVDS Input Only)	31	RN_IO31 (LVDS Input Only)	32
16	RP_IO32	33	RN_IO33	34
17	RP_IO34	35	RN_IO35	36
18	RP_IO36	37	RN_IO37	38
19	RP_IO38	39	RN_IO39_VREF	40
20	RP_IO40	41	RN_IO41	42
21	RP_IO42	43	RN_IO43	44
22	RP_IO44	45	RN_IO45	46
23	RP_IO46 (LVDS Input Only)	47	RN_IO47 (LVDS Input Only)	48
24	RP_IO48	49	RN_IO49	50
25	RP_IO50	51	RN_IO51	52
26	RP_IO52	53	RN_IO53	54
27	RP_IO54	55	RN_IO55_VREF	56
28	RP_IO56	57	RN_IO57	58
29	RP_IO58	59	RN_IO59	60
30	RP_IO60	61	RN_IO61	62
31	RP_IO62	63	RN_IO63	64

This connector is a 64-pin female receptacle header (AMP 120527-1 or equivalent) which mates to the male connector on the carrier/CPU board (AMP 120521-1 or equivalent).

Non-Isolation Considerations

The board is non-isolated, since there is electrical continuity between the logic and field I/O grounds. As such, the field I/O connections are not isolated from the system. Care should be taken in designing installations without isolation to avoid noise pickup and ground loops caused by multiple ground connections.

Standalone Operation

Powering the PMC-LX/SX as an independent board is possible using the J7 board through holes. As an independent standalone board the PMC-LX/SX board would not be plugged into a PMC slot. It must, in this case, receive power though the J7 contact holes and provide passive resistive pullups on all PCI bus signals required to be pulled up by the system card. Note, the board could be damaged if the required pull-up resistors are not used. To independently power the board, the required +5 volt and +3.3 volt power supplies must be provided via the J7 contact holes. The holes have 29 mil openings with 60 mil pads. The location of the J7 contact holes on the board are shown in diagram 4502-053 at the end of this manual.

3.0 PROGRAMMING INFORMATION

This Section provides the specific information necessary to program and operate the board.

GETTING STARTED

1. The PMC LX/SX board is shipped with Xilinx FPGA code stored in flash memory. Upon power-up the PMC LX/SX will automatically configure the FPGA with the example design code stored in flash. As a first step become familiar with the PMC LX/SX, as suppplied by Acromag. The board will perform all the functions of the example design.

   The Example Design Memory Map section gives a description of the I/O operations performed by the example design. It will allow testing of digital I/O, interrupts, read/write of dual port SRAM, read/write of double data rate SDRAM, and testing of both DMA channels. It is strongly recommended that you become familiar with the board features by using the example design as provided by Acromag.

   Do not attempt to reconfigure the flash memory until after you have tested and become familiar with the PMC LX/SX as provided in the example design.

2. After you are familiar with the PMC LX/SX and have tested it using the example design, you can move on to step 2. Here you will modify the example design VHDL code slightly. It is recommended that you test this modified example design using the reconfiguration direct method. It is not recommended that the flash be overwritten until you have tested your code. The reconfigure direct method will allow programming of the FPGA directly from the PCI bus. If for some reason the PMC LX/SX does not perform as expected, you can power the PMC LX/SX down. Upon power-up, the example design provided by Acromag will again be loaded into the FPGA.

The document, Using the PMC LX/SX Engineer Design Kit, provided in the engineering design kit will guide you through the steps required to modify the example design for your custom application.

See the Direct PCI bus to Xilinx Configuration section for a description of the steps required to perform reconfiguration directly from the PCI bus. The registers provided in the FPGA Programming Memory Map are used to implement a direct reconfiguration.

3. After you have thoroughly tested your customized FPGA design, you can erase the flash and write your code to flash. Once the flash is erased you will not be able to go back to the example design by simply powering down and restarting the board.

   See the Flash Configuration section for a description of the steps required to write new design code to the flash device. The registers provided in the FPGA Programming Memory Map are used to implement a flash erase and reprogram operations.

PCI Configuration Address Space

This board is a PCI Specification version 2.2 compliant PCI bus master/target board.

The PCI bus is defined to address three distinct address spaces: I/O, memory, and configuration space. This board can be accessed via the PCI bus I/O, memory, and configuration spaces.

The card's configuration registers are initialized by system software at power-up to configure the card. The board is a Plug-and-Play PCI card. As a Plug-and-Play card the board's base address and system interrupt request line are not selected via jumpers but are assigned by system software upon power-up via the configuration registers. A PCI bus configuration access is used to read/write the PCI card's configuration registers.

When the computer is first powered-up, the computer's system configuration software scans the PCI bus to determine what PCI devices are present. The software also determines the configuration requirements of the PCI card.

The system software accesses the configuration registers to determine how many blocks of memory space the module requires. It then programs the board's configuration registers with the unique memory base address.

The configuration registers are also used to indicate that the board requires an interrupt request line. The system software then programs the configuration registers with the interrupt request line assigned to the board.

Since this board is relocatable and not fixed in address space, its device driver must use the mapping information stored in the board's Configuration Space registers to determine where the board is mapped in memory space and which interrupt line will be used.

CONFIGURATION REGISTERS

The PCI specification requires software driven initialization and configuration via the Configuration Address space. This board provides 512 bytes of configuration registers for this purpose. It contains the configuration registers, shown in Table 3.1, to facilitate Plug-and-Play compatibility.

The Configuration Registers are accessed via the Configuration Address and Data Ports. The most important Configuration Registers are the Base Address Registers and the Interrupt Line Register which must be read to determine the base address assigned to the board and the interrupt request line that goes active on a board interrupt request.

Table 3.1 *Configuration Registers*

Reg. Num.	D31 D24	D23 D16	D15 D8	D7 D0
0	Device ID = 0x4C40(LX40), 0x4C60 (LX60), 0x5335 (SX35)		Vendor ID= 16D5	
1	Status		Command	
2	Class Code=118000			Rev ID=00
3	BIST	Header	Latency	Cache
4	32-bit Memory Base Address for Memory Accesses to Local, Runtime, DMA, and Messaging Queue Registers (PCIBAR0)			
5	PCI Base Address for I/O Accesses to Local, Runtime, DMA, and Messaging Queue Registers (PCIBAR1)			
6	32-bit Memory Base Address for Memory Accesses to Local Address Space 0, 2M Space, FPGA Space (PCIBAR2)			
7 : 10	Not Used			
11	Subsystem ID 0x4C40(LX40), 0x4C60 (LX60), 0x5335 (SX35)		Subsystem Vendor ID=16D5	
12	Not Used			
13,14	Reserved			
15	Max_Lat	Min_Gnt	Inter. Pin	Inter. Line

This board is allocated memory space address (PCIBAR0) to access the PCI9656 runtime, DMA, and messaging queue registers. The PCI9656 decodes 512 bytes for these memory space registers. These registers can also be accessed by an I/O cycle, with the PCI bus address matching the I/O Base Address (PCIBAR1).

In addition, this board is allocated a 2M byte block of memory (PCIBAR2) that is addressable in the PCI bus memory space to control the board's many functions included in the Virtex 4 FPGA.

FPGA PROGRAMMING MEMORY MAP

The memory space address map used to program the FPGA and flash device is shown in Table 3.2. Note that the base address for the board (PCIBAR2) in memory space must be added to the addresses shown to properly access these registers. Register accesses as 32, 16, and 8-bit in memory space are permitted.

Table 3.2: FPGA Programming Memory Map

Base Addr+	D31 D08	D07 D00	Base Addr+
0003	Not Used[1]	Configuration Status Register	**0000**
0007	Not Used[1]	Configuration Control Register	**0004**
000B	Not Used[1]	Configuration Data	**0008**
000F	Not Used[1]	Flash Status 1 Register	**000C**
0013	Not Used[1]	Flash Status 2 Register	**0010**
0017	Not Used[1]	Flash Read	**0014**
001B	Not Used[1]	Flash Reset	**0018**
001F	Not Used[1]	Flash Start Write	**001C**
0023	Not Used[1]	Flash Erase Sector	**0020**
0027	Not Used[1]	Flash Erase Chip	**0024**
002B	Not Used[1]	Flash Data Register	**0028**
002F	Not Used[1]	Flash Address 7->0	**002C**
0033	Not Used[1]	Flash Address 15->8	**0030**
0037	Not Used[1]	Flash Address 22->16	**0034**
003B	Not Used[1]		**0038**
↓	Not Used[1]		↓
7FFF	Not Used[1]		**7FFC**

1. The board will return 0 for all addresses that are "Not Used".

Base Addr+	D31 D16	D15 D00	Base Addr+
8003	Software Reset and Status Register		8000
8007	Mezzanine Module Memory Space		8004
↓			↓
802B			8028
802F	Rear I/O Connector Read Register		802C
8033	Rear I/O Connector Write Register		8030
8037	DMA Control Register		8034
803B	DMA Transfer Size Channel 0		8038
803F	DMA Transfer Size Channel 1		803C
8043	DP-SRAM Control Register		8040
8047	DP-SRAM Address Register		8044
804B	DMA Channel 0 Threshold Register (DP-SRAM)		8048
804F	DMA Channel 1 Threshold Register (DP-SRAM)		804C
8053	Address Reset Register 0 (DP-SRAM)		8050
8057	Address Reset Register 1 (DP-SRAM)		8054
805B	DP-SRAM Read Register		8058
805F	DDR-SDRAM Control Register		805C
8063	DDR-SDRAM Address Register		8060
8067 806B 806F 8073	DDR-SDRAM Read Registers D0 D1 D2 D3		8064 8068 806C 8070
8077 807B 807F 8083	DDR-SDRAM Write Registers D0 D1 D2 D3		8074 8078 807C 8080
8087	DDR-SDRAM Mask Register		8084
808B→ FFFFF	Not Used[1]		8088→ FFFFC
100003 ↓ 1FFFFF	Static RAM Memory		100000 ↓ 1FFFFC

EXAMPLE DESIGN MEMORY MAP

Table 3.2: Example Design Memory Map

1. The board will return 0 for all addresses that are "Not Used".

This memory map reflects byte accesses using the "Little Endian" byte ordering format. Little Endian uses even-byte addresses to store the low-order byte. The Intel x86 family of microprocessors uses "Little Endian" byte ordering. In Big Endian, the lower-order byte is stored at odd-byte addresses. If using a Power PC platform and performing DMA transfers of DP-SRAM data, you will need to switch the PLX PCI9656 to the Big Endian convention. The PLX chip provides a Big/Little Endian descriptor to control Big/Little Endian data formats. See the PLX PCI9656 DMA Setup Registers section of this manual or the PCI9656 user manual.

Flash Configuration

The LX/SX board uses a flash configuration device to store programming information for the Xilinx FPGA. The flash configuration device and FPGA are hardwired together so that during power-up the contents of the configuration device are downloaded to the FPGA. The flash configuration data can be reprogrammed using the PCI bus interface. The following is the general procedure for reprogramming the flash memory and reconfiguration of the Xilinx FPGA:

1) Disable auto-configuration by setting bit-0 (Stop Configuration) of the Configuration Control register to logic high.
2) Clear the Xilinx FPGA of its previous configuration by setting the Configuration Control register bit-2 to logic high. Software must also keep bit-0 set to a logic high.
3) Read INIT as logic high (Bit-1 of Configuration Status register) before programming is initiated.
4) Verify Flash Chip is not busy by reading bit-7 as logic 0 of the Flash Status 2 register at base address plus 10H before starting a new Flash operation.
5) Erase the current flash contents by using the Flash Erase Sector method. Flash erase sectors are implemented by setting bit-0 of the Flash Erase Sector register to logic high. There are 128 flash sectors, which are addressed via the most significant seven flash address lines. The most significant seven flash address lines are set via the Flash Address 22-16 register at base address plus 34H. Issuing a Flash Erase Sector command will erase the contents of the flash chip only in the sector specified.
6) Verify Flash Chip is not busy by reading bit-7 as logic 0 of the Flash Status 2 register at base address plus 10H before going to the next step.
7) Download the Configuration file to the flash configuration chip via the PCI bus.
   i) Write the byte to be sent to the Flash Data register at base address plus 28H.
   ii) Write the address of the Flash Chip to receive the new data byte to the Flash Address registers at base address plus 2CH, 30H, and 34H. Issuing a Flash Start Write will automatically increment this address after the prior Flash Write has been completed. Thus, the address will not need to be set prior to issuing the next Flash Start Write. The first byte of the configuration file should be written to address 0 of the Flash Chip. The Flash Start Write operation will take 9μ seconds to complete.
   iii) Issue a Flash Start Write command to the Flash Chip by writing logic 1 to bit-0 of base address plus 1CH.

iv) Verify the Flash Chip is not busy by reading bit-7 as logic 0 of the Flash Status 2 register at base address plus 10H before going back to step i to write the next byte.

8) Enable auto-configuration by setting bit-0 (Stop Configuration) of the Configuration Control register to logic low.
9) Verify configuration complete by reading DONE (bit-0 of Configuration Status Register) as logic high.
10) Thereafter, at power-up the configuration file will automatically be loaded into the FPGA.

Direct PCI bus to Xilinx Configuration

Configuration of the Xilinx FPGA can be implemented directly from the PCI bus. The following is the general procedure for re-configuration of the Xilinx FPGA via the PCI bus:

1) Disable auto-configuration by setting bit-0 (Stop Configuration) of the Configuration Control register to logic high.
2) Clear the Xilinx FPGA of its previous configuration by setting the Configuration Control register bit-2 to logic high.
3) Read INIT as logic high (Bit-1 of Configuration Status register) before programming is initiated.
4) Download the Configuration file directly to the Xilinx FPGA by writing to the Configuration Data register. The entire configuration file must be written to the Xilinx FPGA one byte at a time to the Configuration Data register at base address plus 08H.
5) Verify configuration complete by reading DONE (bit-0 of Configuration Status Register) as logic high. DONE is expected to be logic high immediately after the last byte of the configuration file is written to the Xilinx FPGA.
6) At each power-up the configuration file will need to be reloaded into the FPGA.

Configuration Status Register (Read Only) – (PCIBAR2 + 0000H)

Configuration Status Register

This read only register reflects the status of configuration complete and Xilinx configuration clear bits. This Configuration Status register is read at base address plus 0H.

Table 3.3: *Configuration Status Register*

Bit(s)	FUNCTION	
0	DONE:	
	0	Xilinx FPGA is not configured
	1	Xilinx FPGA configuration is complete
1	INIT:	
	0	INIT is held low until the Xilinx is clear of its current configuration
	1	INIT transitions high when the clearing of the current Xilinx configuration is complete
2 to 7	Not Used (bits are read as logic "0")	

CONFIGURATION CONTROL REGISTERS

Configuration Control (Read/Write) – (PCIBAR2 + 04H)

This read/write register is used to stop Xilinx configuration, clear Xilinx configuration memory, and set Local Bus Hold control. This Configuration Control register is accessed at base address plus 04H.

Table 3.4: *Configuration Control Register*

Bit(s)	FUNCTION	
0	Stop Xilinx Configuration:	
	0	Enable Xilinx FPGA configuration
	1	Stop Xilinx FPGA configuration
1	Not Used (bit is read as logic "0")	
2	Clear Current Xilinx Configuration:	
	0	Logic low has no effect.
	1	Logic high resets the Xilinx configuration logic. Re-configuration can begin after INIT transitions high.
3 to 6	Not Used (bits are read as logic "0")	
7	Local Bus Hold Control	
	0	CPLD controls generation of Local Hold Acknowledge. The CPLD logic will always grant control of the local bus to the PCI9656 device.
	1	Xilinx FPGA controls generation of the Local Hold Acknowledge signal.

Configuration Data (Write Only) – (PCIBAR2 + 08H)

This write only register is used to write Xilinx configuration data directly to the Xilinx FPGA from the PCI bus. The Configuration Data register is accessed at base address plus 08H. The entire configuration file must be written to the Xilinx FPGA one byte at a time. Configuration complete is verified by reading DONE (bit-0 of Configuration Status Register) as logic high.

A write to the Configuration Data register while auto-configuration from Flash is active will cause the Xilinx configuration to fail. Auto-configuration is stopped by writing logic 1 to bit-0 of the Configuration Control register at base address plus 04H.

The Xilinx FPGA should also be cleared of its current configuration prior to loading of a new configuration file. The FPGA is cleared of its current configuration by writing logic 1 to bit-2 at address plus 04H.

Flash Status 1 (Read Only) – (PCIBAR2 + 0CH)

This read only register is used to read the DQ5 status of the flash chip. The Flash Status 1 register is at base address plus 0CH.

Bit(s)	FUNCTION	
0 to 4	Not Used (bits are read as logic "1 or 0")	
5	DQ5:	
	0	Chip enabled for reading array data.
	1	The system must issue the Flash Reset command to re-enable the device for reading array data if DQ5 goes high. DQ5 can go high during a Flash Start Write, Flash Erase Chip, or Flash Erase Sector operation.
6 and 7	Not Used (bits are read as logic "1 or 0")	

Table 3.5: *Flash Status 1 Register*

Flash Status 2 (Read Only) – (PCIBAR2 + 10H)

This read only register is used to read the ready or busy status of the flash chip. The Flash Status 2 register is at base address plus 10H. The system must first verify that that Flash Chip is not busy before executing a new Flash command. The Flash Chip is busy if bit-7 of this register is set to logic 1. The Flash will always be Busy while bit-0 of the Configuration Control register is set to logic "0".

Bit(s)	FUNCTION	
0 to 6	Not Used (bits are read as logic "0")	
7	Busy / Ready~ Set bit-0 of the Configuration Control register to logic "1" before monitoring this busy bit.	
	0	Flash Chip is Ready
	1	Flash Chip is Busy

Table 3.6: *Flash Status 2 Register*

Flash Read (Read Only) – (PCIBAR2 + 14H)

A Flash Read command is executed by reading this register at base address plus 14H. Prior to issue of a Flash Read the Flash Address registers must be set with the desired address to be read. See the Flash Address registers at base address plus 2CH, 30H, and 34H.

The system must issue the Flash Reset command to re-enable the device for reading array data if DQ5 goes high. DQ5 can go high during a Flash Start Write, Flash Erase Chip, or Flash Erase Sector operation. DQ5 can be monitored via the Flash Status 1 register at base address plus 0CH.

Flash Reset (Write Only) – (PCIBAR2 + 18H)

This write only register is used to initiate a reset of the flash chip. A Flash Reset command is executed by writing logic 1 to bit-0 of this register at base address plus 18H. Writing the flash reset command resets the chip to reading data mode. Flash reset can be useful when busy is held active. Busy can be held active when write of a logic high is executed to a bit that is currently a logic low. The only way to set bits to logic high is by way of a Flash chip erase.

FLASH CONTROL REGISTERS

Flash Start Write (Write Only) – (PCIBAR2 + 1CH)

This write only register is used to initiate the write of a byte to the flash chip. A Flash Start Write command is executed by writing logic 1 to bit-0 of this register at base address plus 1CH. Prior to issue of a Flash Start Write the Flash Data and Address registers must be set with the desired data and address to be written. See the Flash Data and Address registers at base address plus 28H, 2CH, 30H, and 34H.

Issue of a Flash Start Write will automatically increment this address after the issued Flash Write has completed. Thus, the address will not need to be set prior to issue of the next Flash Start Write if consecutive addresses are to be written.

Flash Erase Sector (Write Only) – (PCIBAR2 + 20H)

A Flash Erase Sector command is executed by writing logic 1 to bit-0 of this register at base address plus 20H. Verify that the Flash Chip is not busy from a previous operation before beginning a new operation. This is accomplished by reading bit-7 as logic 0 of the Flash Status 2 register.

There are 128 flash sectors, which are addressed via the most significant seven flash address lines. The most significant seven flash address lines are set via the Flash Address 22-16 register at base address plus 34H. Issue of a Flash Erase Sector command will erase the contents of the flash chip only in the sector specified.

A flash bit cannot be programmed from logic 0 back to logic 1. Only an erase chip operation can convert logic 0 back to logic 1. **Prior to reprogramming of the flash chip a Flash Erase Chip or Flash Erase Sector command must be performed.**

The system can determine the status of the erase operation by reading the Flash Ready/Busy status. Bit-7 of the Flash Status 2 register, at base address plus 10H, will read as logic 0 when chip erase is completed.

Any other flash commands written to the flash chip during execution of the flash erase sector operation are ignored. Note that a hardware reset during the erase sector operation will immediately terminate the operation.

Flash Erase Chip (Write Only) – (PCIBAR2 + 24H)

This write only register is used to erase the entire contents of the flash chip. A flash bit cannot be programmed from logic 0 back to logic 1. Only an erase chip operation can convert logic 0 back to logic 1. **Prior to reprogramming of the flash chip a Flash Erase Chip command must be performed.**

A Flash Erase Chip command is executed by writing logic 1 to bit-0 of this register at base address plus 24H. Verify that the Flash Chip is not busy from a previous operation before beginning a new operation. This is accomplished by reading bit-7 as logic 0 of the Flash Status 2 register.

The system can determine the status of the erase operation by reading the Flash Ready/Busy status. Bit-7 of the Flash Status 2 register, at base address plus 10H, will read as logic 0 when chip erase is completed.

Any other flash commands written to the flash chip during execution of the flash erase chip operation will be ignored. Note that a hardware reset during the chip erase operation will immediately terminate the operation.

Flash Data Register (Read/Write) – (PCIBAR2 + 28H)

This read/write register holds the data byte which is sent to the flash chip upon issue of a Flash Start Write command.

Although only the least significant 8 bits of this register are used, reading or writing this register is possible via 32-bit, 16-bit or 8-bit data transfer.

Flash Address 7->0 (Read/Write) – (PCIBAR2 + 2CH)

This read/write register holds the least significant byte of the address to which the flash chip is written upon issue of a Flash Start Write command.

Although only the least significant 8 bits of this register are used, reading or writing this register is possible via 32-bit, 16-bit or 8-bit data transfer.

Flash Address 15->8 (Read/Write) – (PCIBAR2 + 30H)

This read/write register sets bits 15 to 8 of the address to which the flash chip is written upon issue of a Flash Start Write command.

Although only the least significant 8 bits of this register are used, reading or writing this register is possible via 32-bit, 16-bit or 8-bit data transfer.

Flash Address 22->16 (Read/Write) – (PCIBAR2 + 34H)

This read/write register sets bits 22 to 16 of the address to which the flash chip is written upon issue of a Flash Start Write command. The most significant bit of this register is not used.

Although only the least significant 7 bits of this register are used, reading or writing this register is possible via 32-bit, 16-bit or 8-bit data transfer.

RESET REGISTER

Software Reset and Status Register (Read/Write) – (PCIBAR2 + 8000H)

This read/write register is used to Software reset the board, monitor the status of board interrupts, and monitor the temperature of the Xilinx virtex-4 FPGA.

Reading or writing to this register is possible via 32-bit, 16-bit or 8-bit data transfers.

Table 3.7: *Software Reset and Status Register*

1. *All bits labeled "Not Used" will return logic "0" when read.*

BIT	FUNCTION	
7-0	Mezzanine interrupt status is identified via data bits 0 to 7. Read of a "1" indicates that an interrupt is pending for the corresponding data bit. A pending interrupt will remain active until disabled via the mezzanine interrupt control registers.	
	Logic "0"	Interrupt Not Pending
	Logic "1"	Interrupt Pending
12-8[1]	Not Used[1]	
15-13	Mezzanine Identification Code: "001" for all digital I/O mezzanine boards	
28-16	A digital temperature sensor monitors the temperature of the Xilinx Virtex-4 FPGA. The digital temperature sensor has an accuracy of +/-2.4°C. The data is available as a 13-bit two's complement value. The 13-bit data value provides a resolution of 0.0625°C. Multiply the signed 3-bit data value by 0.0625°C to obtain the FPGA die temperature.	
	Temperature	D28-D16
	150°C	0,1001,0110,0000
	125°C	0,0111,1101,0000
	25°C	0,0001,1001,0000
	0.0625°C	0,0000,0000,0001
	0°C	0,0000,0000,0000
	-0.0625°C	1,1111,1111,1111
	-25°C	1,1110,0111,0000
	-55°C	1,1100,1001,0000
30-29	Not Used[1]	
Bit-31	The most significant bit of this register when set to a logic "1" will issue a software reset.	
	Logic "0"	No Operation
	Logic "1"	Software Reset issued to Xilinx Virtex-4 FPGA

DMA INTERRUPT REGISTERS

Interrupts must be enabled via the PCI9656 control registers and the Mezzanine Interrupt Enable register, in order to generate interrupts. The PCI9656 Interrupt Control/Status register at PCIBAR0 base address + offset 68H must have bits 8 and 11 set to logic high in order for interrupts to occur.

DMA interrupts must be enabled and controlled through the PCI9656 registers. The PCI9656 Interrupt Control/Status register at PCIBAR0 base address + offset 68H must have bits 18 and 19 set to a logic high in order for DMA interrupts to occur on DMA channels 0 and 1, respectively.

DMA transfers are configured and controlled via PCI9656 DMA registers. The PCI9656 DMA registers are at PCIBAR0 base address + offset 80H to B8H. These registers control the transfer direction, size, source address, and destination address for DMA channels 0 and 1.

Rear I/O Connector Read Register (Read Only) - (PCIBAR2 + 802CH)

REAR Read REGISTER

The Rear I/O Connector Read Register is used to read the LVDS input status of 16 channels. This example design has 16 channels, identified in Table 3.8, programmed as LVDS input only channels. Table 2.1 shows each channel and it's corresponding P4 connector pin assignment.

This Rear I/O Connector Read register is a read only register and writing to this register has no effect on the LVDS input channels. Reading from this register is possible via 32-bit, 16-bit or 8-bit data transfers.

Rear I/O Connector Write Register (Read/Write) - (PCIBAR2 + 8030H)

REAR Write REGISTER

The Rear I/O Connector Write Register is used to set 16 LVDS output channels. This example design has 16 channels, identified in Table 3.8, fixed as LVDS output only channels. Table 2.1 shows the P4 connector pins and their corresponding channel identifiers.

This Rear I/O Connector Write register is written to set the LVDS output channels and can also be read to verify the output channel settings. Reading from this register is possible via 32-bit, 16-bit or 8-bit data transfers.

Rear Connector Register Bit	Rear Connector Write Register Output Channels	Rear Connector Read Register Input Channels
0	1	0
1	2	3
2	4	5
3	6	7
4	8	9
5	10	11
6	12	13
7	14	15
8	16	17
9	18	19
10	20	21
11	22	23
12	24	25
13	26	27
14	28	29
15	30	31
16-31	Not Used[1]	Not Used[1]

Table 3.8: *Rear I/O Registers*

Column 1 identifies the write data bit that drives the output channel listed in column 2. Column 1 also identifies the read data bit that returns the input channel listed in column 3. For example data bit 0 drives output channel 1 when written and returns channel 1 register setting when read.

1. *All bits labeled "Not Used" will return logic "0" when read.*

DMA REGISTERS

DMA Control Register (Read/Write) - (PCIBAR2 + 8034H)

The DMA Control Register is used to request a DMA Demand mode transfer. The transfer must include the Static RAM Memory as either the source or the destination.

Bit-0 is used to request a DMA channel 0 transfer while bit-1 is used to request a channel 1 DMA transfer. The bit must be set to logic high to request a transfer. Once set, the bit will remain set until the DMA transfer has completed. If both bits are set simultaneously, the channel 0 DMA transfer will be implemented first followed by the channel 1 DMA transfer.

DMA REGISTERS

PLX PCI9656 DMA Setup Registers

The size of the DMA transfer must be set in the DMA Transfer Size register corresponding to the channel handling the transfer. See the description of the DMA Transfer Size registers in the following paragraphs. In addition, the DMA transfer size, direction, source and destination must be set in the PCI9656 DMA control registers. The PCI9656 DMA registers are at PCIBAR0 base address + offset 80H to B8H. Specifically, the registers and bits must be assigned as shown in the table below when performing a DMA channel for proper operation.

Table 3.9: *PCI9656 DMA Register Setup Values*

Big Endian Mode will be required for Power PC system. Note that any registers/bits not mentioned may remain at the default value.

This is not an all-inclusive list. Other DMA features are available that are not listed. See the PCI9656 manual for more details.

DMA transfers must start aligned to a Lword boundary.

PCIBAR0 Base Addr+	Bit(s)	Logic Value	Description
0CH	6	0	DMA Channel 1 Little Endian Mode
		1	DMA Channel 1 Big Endian Mode
	7	0	DMA Channel 0 Little Endian Mode
		1	DMA Channel 0 Big Endian Mode
80H (Chan. 0)/ 94H (Chan. 1)	1:0	11	DMA Local Bus Data Width (32 bits)
	6	1	DMA READY# Input Enable
	7	0	DMA Burs-4 Mode Enable
	8	1	DMA Local Burst Enable
	11	0	DMA Local Address is Incremented
	12	1	DMA Demand Mode Enable
	15	0	DMA Slow Terminate mode
84H (Chan. 0) / **98H (Chan. 1)**	31:0	User defined	PCI DMA starting Address
88H (Chan. 0)/ 9CH (Chan. 1)	31:0	User defined	Local DMA starting Address
8CH (Chan 0) / **A0H (Chan. 1)**	22:0	User defined	DMA Transfer Size
90H (Chan. 0) / A4H (Chan. 1)	3	0	DMA Transfer PCI -> LX/SX Board
		1	DMA Transfer LX/SX Board –> PCI
A8H (Chan 0) / A9H (Chan. 1)	0	1	DMA Channel Enable
	1	1	∗DMA Channel Start

∗Every time you need to perform a new demand mode DMA transfer, you must reset the DMA Channel Start bit.

See the PCI9656 user's manual and Acromag's software source code which provides an example for further details.

The DMA Control and DMA Transfer Size registers are only used to initiate DMA Demand Mode transfers. These registers are used to illustrate DMA Demand Mode data transfers. Writing to these registers is possible via 32-bit, 16-bit or 8-bit data transfers.

DMA REGISTERS

DMA Transfer Size Registers (Read/Write) - (PCIBAR2 + 8038H and 803CH)

The DMA Transfer Size Register is used to set the size of the DMA Demand mode data transfer that moves data to or from the onboard Static RAM memory. The onboard static RAM has 256K memory locations. As such, the maximum value that can be written to this register is 3FFFFH which corresponds to 2^{18}. Note that the transfer size specifies the number of Lwords transferred. A value of 3FFFFH would specify the move of 256K long words.

The DMA Transfer Size Register at base address + 8038H is used to set the DMA channel 0 data transfer size. The register at base address + 803CH is used to set the DMA channel 1 data transfer size. Writing to these registers is possible via 32-bit, 16-bit or 8-bit data transfers.

DP-SRAM REGISTERS

Dual-Port SRAM Control Register (Read/Write) – (PCIBAR2 + 8040H)

This read/write register is used to control the Dual-Port SRAM including enabling writing, automatic DMA transfer and automatic address reset on DMA thresholds.

The default power-up state of this register is logic low. A reset will set all bits in this register to "0". Reading or writing to this register is possible via 32-bit, 16-bit or 8-bit data transfers.

Table 3.10: *Dual-Port SRAM Control Register*

1. *Bits are not used and will return logic "0" when read.*

2. *All DMA transfer settings in the DMA Registers and on the PLX9656 should be set prior to enabling automatic DMA transfers.*

3. *WARNING: Before enabling Address Reset on DMA Thresholds (bits 3 & 4), verify that the "DMA Ch. 0 Threshold Register" is not equal to the "Address Reset Register 0" and the "DMA Ch. 1 Threshold Register" is not equal to the "Address Reset Register 1." If these registers are equal and automatic reset is enabled an infinite loop will be created within the internal logic of the FPGA.*

BIT	FUNCTION	
0	Rear I/O channels 0 to 15 can be sampled and written to SRAM if enabled via this bit. The SRAM Internal Address register must also be set with the start address at which the data begins filling the SRAM. Writing to the SRAM from the Rear I/O port will automatically be disabled upon reaching the DMA Channel 1 Threshold value. Reset on DMA Thresholds (bit-3 and bit-4 of this register) should not be enabled.	
	Logic "0"	Disable Rear I/O write to SRAM
	Logic "1"	Enable Rear I/O write to SRAM
1	If enabled via this bit a DMA channel 0 request will be issued when the internal address counter is equal to the DMA Channel 0 Threshold Register. This will have the same effect as writing a 1 to bit 0 of the DMA Control Register. See Synchronous DP-SRAM in Section 4.0 for further details on using this feature.[2]	
	Logic "0"	Disable Auto DMA Request Channel 0
	Logic "1"	Enable Auto DMA Request Channel 0
2	If enabled via this bit a DMA Channel 1 request will be issued when the internal address counter is equal to the DMA Channel 1 Threshold Register. This will have the same effect as writing a 1 to bit 1 of the DMA Control Register.[2]	
	Logic "0"	Disable Auto DMA Request Channel 1
	Logic "1"	Enable Auto DMA Request Channel 1
3	If enabled via this bit the Internal Address Counter will be loaded with the value in Address Reset Register 0 when the counter is equal to the DMA Channel 0 Threshold Register. See the Address Reset Register description for further details. The DMA does not have to be enabled to use this feature.[3]	
	Logic "0"	Disable Add. Reset on DMA Ch. 0 Threshold
	Logic "1"	Enable Add. Reset on DMA Ch. 0 Threshold
4	If enabled via this bit the Internal Address Counter will be loaded with the value in Address Reset Register 1 when the counter is equal to the DMA Channel 1 Threshold Register. See the Address Reset Register description for further details. The DMA does not have to be enabled to use this feature.[3]	
	Logic "0"	Disable Add. Reset on DMA Ch. 1 Threshold
	Logic "1"	Enable Add. Reset on DMA Ch. 1 Threshold
5	DMA Burst Control	
	Logic "0"	DMA Performed Without Burst
	Logic "1"	DMA Burst Transfer Enabled
6-15	Not Used[1]	

DP-SRAM REGISTERS

Dual-Port SRAM Internal Address Register (Read/Write) – (PCIBAR2 + 8044H)

The Dual-Port SRAM Internal Address Register is used to view and set the internal SRAM address. The FPGA will only write using 32-bit data transfers allowing for 3FFFFH (2^{18}) unique memory accesses. **Reading** bits 0 to 17 of this register will return the internal SRAM address. *Due to delays during data processing and the PCI transfer the actual internal address may be slightly greater than the read address.* **Writing** to bits 0 through 17 of this register will set the Internal SRAM Address to the provided value. Writing logic '1' to bit 31 of this register or a system reset will cause the Internal SRAM Address to reset to "00000H" (the start of the SRAM memory). **Reading or writing to this register is possible via 32-bit data transfers, only.**

Warning: To guarantee functionality disable DP-SRAM write cycles (via bit 0 of the DP-SRAM Control Registers) before writing to the DP-SRAM Internal Address Register.

Dual-Port SRAM Internal Address Register		
D31	**D30-D18**	**D17-D0**
SRAM Internal Address Reset	Not Used (Read as logic '0')	SRAM Internal Address

Table 3.11: *Dual-Port SRAM Internal Address Register*

Dual-Port SRAM DMA Channel 0/1 Threshold Registers (Read/Write) – (PCIBAR2 + 8048H/ 804CH)

The Dual-Port SRAM DMA Channel 0/1 Threshold Registers are used to initiate an automatic DMA transfer. When the internal address counter is equal the value in the DMA Channel 0 Threshold Register, a Channel 0 DMA request will be initiated. Similarly, when the internal address counter is equal the value in the DMA Channel 1 Threshold Register and there is valid data at that address, a Channel 1 DMA request will be initiated. This feature must be enabled via bits 1 and 2 (for Channels 0 & 1, respectively) of the DP-SRAM Control Register. Note that DMA settings must be set prior to the initiated transfer on both the PCI9656 and in the DMA Registers. A DMA transfer in progress is indicated via bits 0 and 1, for DMA Channels 0 and 1, respectively, in the DMA Control Register. See the DMA Registers section of this manual for further details. **Reading** bits 0 to 17 of either register will return the corresponding DMA Threshold. **Writing** to bits 0 through 17 of this register will set the corresponding DMA Threshold to the provided value. **Reading or writing to this register is possible via 32-bit data transfers only.**

Note: A SRAM DMA Request will occur only <u>after</u> a data write cycle to the address defined by the DMA Threshold Registers.

Table 3.12: *Dual-Port DMA Threshold Registers*

Dual-Port SRAM DMA Channel 0/1 Threshold Registers			
Register	**D31-D18**	**D17-D0**	**Reset Value**
DMA Channel 0 Threshold Reg.	Not Used (Read as logic '0')	Channel 0 Threshold Address	1FFFFH
DMA Channel 1 Threshold Reg.	Not Used (Read as logic '0')	Channel 1 Threshold Address	3FFFFH

DP-SRAM REGISTERS

WARNING: The "DMA Ch. 0 Threshold Register" must not equal the "Address Reset Register 0" and the "DMA Ch. 1 Threshold Register" must not equal the "Address Reset Register 1." If these registers are equal and the address reset is enabled via the DP-SRAM Control Register an infinite loop will be created within the internal logic of the FPGA.

Dual-Port SRAM Address Reset Registers 0/1 (Read/Write) – (PCIBAR2 + 8050H/ 8054H)

The Dual-Port SRAM Address Reset Registers are used to reset the internal address counter to a user-defined value immediately upon reaching the DMA Threshold value. For example, after a SRAM write cycle where the internal address counter is equal to the value defined in the DMA Channel 0 Threshold Register, the internal address counter will then be loaded with the value defined in the Address Reset Register 0. Similarly, after a SRAM write cycle where the internal address counter is equal to the value defined in the DMA Channel 1 Threshold Register, the internal address counter will then be loaded with the value defined in the Address Reset Register 1. This allows for the internal address counter to be changed without any interruption in the transfer of data from the front connector input to the DP-SRAM. This feature must be enabled via bits 3 and 4 (for Channel 0 & 1 thresholds, respectively) of the DP-SRAM Control Register. Note that the DMA transfers *do not* have to be enabled for this feature to function. **Reading** bits 0 to 17 of either register will return the corresponding internal address reset value. **Writing** to bits 0 through 17 of this register will set the corresponding internal address reset register to the provided value. Bits 18 to 31 are not used and will return logic '0' when read. A system reset will cause these registers to reset to "00000H".

Dual-Port SRAM Read Register (Read Only) – (PCIBAR2 + 8058H)

The Dual-Port SRAM Read Register is provided for production testing of the LX/SX product. This port can be used to verify the integrity of the SRAM port linked to the FPGA. Reading this register is only possible in 32-bit transfers. The address in SRAM read is initialized by the Dual Port SRAM Internal Address register at base address + 8044H. With each additional read the address is automatically incremented.

DDR SDRAM Control Register (Read/Write) –(PCIBAR2 + 805CH)

DDR-SDRAM REGISTERS

This read/write register is used to control burst read or write to DDR SDRAM. In addition a DDR-SDRAM error status bit is available in this register. The DDR-SDRAM is set for a burst length of four and will require the DDR-SDRAM Write register to be preloaded with four 32-bit data values prior to issuing of the write operation. The DDR-SDRAM Read register will contain four 32-bit data values upon issue of a read operation. For either a read or write the DDR-SDRAM Address register must be written with the desired command and address location for the access.

BIT	FUNCTION	
0	Start DDR-SDRAM write operation. The DDR-SDRAM Write and Mask registers must first be written with the desired data that are burst out to the DDR-SDRAM. In addition, the DDR-SDRAM Address register must be written with the write address and command prior to setting this bit.	
	Logic "0"	No operation performed
	Logic "1"	Write Transfer Performed
1	Start DDR-SDRAM read operation. The Address Register must be written with the start address location and the read command prior to setting this bit. The DDR-SDRAM Read registers are filled with four data words that are burst out from the DDR-SDRAM.	
	Logic "0"	No operation performed
	Logic "1"	Read Transfer Performed
2	Start DDR-SDRAM command operation. A precharge or active command loaded into the Address Register can be initiated by setting this bit.	
	Logic "0"	No operation performed
	Logic "1"	Command Performed
3-25	Not Used[1]	
26	WDF_Almost_Full: DDR Write data FIFO is almost full when this bit is Logic "1".	
27	AF_Almost_Full: DDR Address FIFO is almost full when this bit is Logic "1".	
30,29,28	DDR burst length	
	Logic "001"	Burst Length = 2
	Logic "010"	Burst Length = 4
	Logic "011"	Burst Length = 8
Bit-31	DDR-SDRAM Error Status	
	Logic "0"	No Error
	Logic "1"	Error Condition

Table 3.13: *DDR-SDRAM Control Register*

1. *All bits labeled "Not Used" will return logic "0" when read.*

DDR SDRAM Address Register (Read/Write) –(PCIBAR2 + 8060H)

This read/write register is used to set the DDR-SDRAM column address, row address, bank, chip select, and command. This Address register must be written prior to issuing of a DDR-SDRAM read or write burst transfer via bits 0 or 1 of the DDR-SDRAM Control register.

Table 3.14: *DDR-SDRAM Address/Command Register*

BIT	FUNCTION	
A9-A0	DDR-SDRAM Column address is written to these bits. A column address is required when a read or write command is present on bits 29 to 31. There are 1024 unique columns.	
A10	A10 is used for precharge control with a read or write command. If A10 is logic high, auto precharge is selected and the row being accessed will be precharged at the end of the write. If A10 is logic low, the row will remain open for subsequent accesses.	
	Logic "0"	Disables auto precharge
	Logic "1"	Enables auto precharge
	A10 will also determine whether one or all banks are precharged when a precharge command is issued.	
	Logic "0"	Single precharge BA1 and BA0 (A25,A24) determine which bank is precharged.
	Logic "1"	All Banks are precharged
A23-A11	DDR-SDRAM Row address is written to these bits. There are 8192 unique rows,	
A25,A24	DDR-SDRAM Bank address (BA1, BA0) is written to these bits. The DDR-SDRAM has four unique banks.	
A26	DDR-SDRAM Chip Select (CS). This bit should always be set to logic High.	
A28,A27[1]	Not Used[1]	

1. *All bits labeled "Not Used" will return logic "0" when read.*

BIT	FUNCTION	
A31-A29	DDR-SDRAM Command	
	Logic "000"	**Load Mode Register** The burst length, burst type, CAS latency, and operating mode are defined in the mode register. These mode control settings are programmed via the mem_interface_top_parameters.vhd parameter file.
	Logic "001"	**Auto Refresh** The DDR is required to be refreshed once every 7.8μs. The Xilinx DDR controller will automatically issue the refresh command. The refresh frequency is based on a 150Mhz DDR clock frequency. If the DDR clock frequency is changed, the max_ref_cnt parameter in the mem_interface_top_parameters.vhd file needs to be changed accordingly. For example at 150MHz max_ref_cnt is "1001000".
	Logic "010"	**Precharge** The precharge command is used to deactivate the open row in a particular bank or an open row in all banks. Address signal A10 is held High during Precharge All Banks. A10 held Low during single bank precharge and is selected by BA1 and BA0 (A25 and A24)
	Logic "011"	**Active** This command activates a row in a bank, allowing any Read or Write to be issued to a bank in the memory array. The value on BA1 and BA0 (A25 and A24) selects the bank, and the address on A23 to A11 selects the row. The row remains active for accesses until a precharge command is issued to that bank. A precharge command must be issued before opening a different row in the same bank. When the controller detects an incoming address that refers to a row in a bank other than the currently opened row, the controller issues a precharge to deactivate the open row and then issues another Active command to the new row.
	Logic "100"	**Write** The write burst access is initiated to the active row. The value on BA1 and BA0 selects the bank address, while the value on A9 to A0 selects the starting column location in the active row. See bit A10 above for precharge description.
	Logic "101"	**Read** The read burst access is initiated to the active row. The value on BA1 and BA0 selects the bank address, while the value on A9 to A0 selects the starting column location in the active row. See bit A10 above for precharge description.
	Logic "111"	**No Operation (NOP)**

Table 3.14 Continued: *DDR-SDRAM Address/Command Register*

DDR SDRAM Read Registers (Read Only) – (PCIBAR2 + 8064H to 8070H)

The four DDR-SDRAM Read registers are read only and hold the last four data values read from the DDR-SDRAM. The DDR-SDRAM is set for a burst of four for the purposes of this design example.

A DDR-SDRAM read is implemented by executing the following steps.

1) Issue a Precharge Command

   a) Write the following value to the SDRAM Address Register at 8060H.

A31-A29	A28,A27	A26	A25,A24	A23-A11	A10	A9-A0
010	X	1	Bank	Row	1	X

   b) Send the precharge command out by setting bit-2 of the SDRAM Control Register at 805CH to a logic '1'.

2) Issue an Active Command

   a) Write the following value to the SDRAM Address Register at 8060H.

A31-A29	A28,A27	A26	A25,A24	A23-A11	A10	A9-A0
011	X	1	Bank	Row	0	X

   b) Send the active command out by setting bit-2 of the SDRAM Control Register at 805CH to a logic '1'.

3) Issue the Read Command

   a) Set the DDR-SDRAM Address register with the start address location and read command. Write the following value to the SDRAM Address Register at 8060H.

A31-A29	A28,A27	A26	A25,A24	A23-A11	A10	A9-A0
101	X	1	Bank	Row	0	Column

   b) Set the start read bit of the DDR-SDRAM Control register at base address + 805C. Set bit-1 of the SDRAM Control Register at 805CH.

The data is read from the SDRAM and moved to the SDRAM Read Registers at 8064H to 8070H. Reading these registers is possible via 32, 16 or 8-bit transfers.

Table 3.15: *DDR-SDRAM Read Registers*

DDR SDRAM Read Registers		
Base Addr+	**D31-D0**	**Base Addr+**
8067	DDR-SDRAM Read Register D0	**8064**
806B	DDR-SDRAM Read Register D1	**8068**
806F	DDR-SDRAM Read Register D2	**806C**
8073	DDR-SDRAM Read Register D3	**8070**

DDR SDRAM Write Registers (Read/Write) – (PCIBAR2 + 8074H to 8080H)

The four DDR-SDRAM Write registers hold four data values that are to be written to the DDR-SDRAM. The DDR-SDRAM is set for a burst of four for the purposes of this design example.

A DDR-SDRAM write is implemented by executing the following steps.

1) Issue a Precharge Command

   a) Write the following value to the SDRAM Address Register at 8060H.

A31-A29	A28,A27	A26	A25,A24	A23-A11	A10	A9-A0
010	X	1	Bank	Row	1	X

   b) Send the precharge command out by setting bit-2 of the SDRAM Control Register at 805CH to a logic '1'.

2) Issue an Active Command

   a) Write the following value to the SDRAM Address Register at 8060H.

A31-A29	A28,A27	A26	A25,A24	A23-A11	A10	A9-A0
011	X	1	Bank	Row	0	X

   b) Send the active command out by setting bit-2 of the SDRAM Control Register at 805CH to a logic '1'.

3) Write the four 32-bit data values that are to be written to the DDR-SDRAM to the registers at base address + 8074H to 8080H.

DDR SDRAM Write Registers		
Base Addr+	**D31-D0**	**Base Addr+**
8077	DDR-SDRAM Write Register D0	**8074**
807B	DDR-SDRAM Write Register D1	**8078**
807F	DDR-SDRAM Write Register D2	**807C**
8083	DDR-SDRAM Write Register D3	**8080**

Table 3.16: *DDR-SDRAM Write Registers*

4) Set the DDR-SDRAM Mask bits as desired at base address + 8084H. A value of 0H would enable all bytes to be written.

5) Issue the Write Command

   a) Set the DDR-SDRAM Address register with the start address location and write command. Write the following value to the SDRAM Address Register at 8060H.

A31-A29	A28,A27	A26	A25,A24	A23-A11	A10	A9-A0
100	X	1	Bank	Row	0	Column

b) Set the start write bit of the DDR-SDRAM Control register at base address + 805C. Set bit-0 of the SDRAM Control Register at 805CH.

DDR SDRAM Mask Register (Read/Write)–(PCIBAR2 + 8084H)

The DDR-SDRAM mask register holds the write mask data bits that accompany the write data as it is written to the DDR-SDRAM. If a given data mask (DM) bit is set low, the corresponding data will be written to memory; if the DM bit is set high, the corresponding data will be ignored, and a write will not be executed to that byte location. The DDR-SDRAM is set for a burst of four for the purposes of this design example.

Table 3.17: *DDR-SDRAM Mask Register*

DDR SDRAM Mask Register Bit	Write Register	Byte Masked
0	D0	Byte 0 (D0 to D7)
1		Byte 1 (D8 to D15)
2		Byte 2 (D16 to D23)
3		Byte 3 (D24 to D31)
4	D1	Byte 0 (D0 to D7)
5		Byte 1 (D8 to D15)
6		Byte 2 (D16 to D23)
7		Byte 3 (D24 to D31)
8	D2	Byte 0 (D0 to D7)
9		Byte 1 (D8 to D15)
10		Byte 2 (D16 to D23)
11		Byte 3 (D24 to D31)
12	D3	Byte 0 (D0 to D7)
13		Byte 1 (D8 to D15)
14		Byte 2 (D16 to D23)
15		Byte 3 (D24 to D31)
16-31	Not Used[1]	

2. *All bits labeled "Not Used" will return logic "0" when read.*

Read or writing this register is possible via 32, 16 or 8-bit transfers.

STATIC RAM MEMORY

Static RAM Memory (Read/Write) – (PCIBAR2 + 100000H to 1FFFFFH)

The Static RAM memory space is used to provide read or write access to on board SRAM memory. The Static RAM device has a 256K x 36-bit memory configuration. Reading or writing to this memory space using DMA access is also only possible in 32-bit transfers.

DUAL PORT MEMORY

A 256K x 36-bit Dual Port synchronous SRAM (DP-SRAM) is provided on the LX/SX board. One port of the SRAM connects directly to the local bus to allow for PCI access. The remaining port connects directly with the FPGA. This design allows for the user to maximize data throughput between the Field I/O's and the controlling processor.

There are two automatic DMA initiators available that will trigger upon a user set threshold. Furthermore, upon a DMA transfer, the internal counter can be reset to a user specified value. See DMA Registers for more information on these operations. These features can be individually controlled through the SRAM Control Registers.

Dual-Port SRAM DMA Transfer Example

In this example external data is written to LX/SX board SRAM. Upon receiving the 131,072 Lwords the DMA demand mode transfer is started. The DMA transfer will move the 131,072 Lword samples to a designated location in system memory.

1. The PCI 9656 DMA Channel 0 registers must first be set to transfer 131,070 Lwords.
   - Write 1043H to PLX DMA Channel 0 Mode register at PCIBAR0 plus 80H.
   - Write the destination start address. This is the location in system memory that the LX/SX SRAM data will be moved. The address is written to the PLX DMA Channel 0 PCI Address register at PCIBAR0 plus 84H.
   - Write the source start address. This is the start address of the LX/SX SRAM. Write PCIBAR2 + 100000H to location PCIBAR0 plus 88H (PLX DMA Channel 0 Local Address register).
   - Write the number of bytes moved by the DMA transfer to the PLX DMA Channel 0 Transfer Size register. Write 80000H (524,288 bytes, decimal) to PCIBAR0 plus 8CH.
   - Setup the DMA transfer direction as LX/SX board to PCI. Write 8H to PCIBAR0 plus 90H (PLX DMA Channel 0 Descriptor Pointer register).
   - Enable the PLX chip for the DMA transfer. Write 1H to PCIBAR0 plus A8H (PLX DMA Channel 0 Command/Status register).
   - Set the PLX chip up to start the DMA transfer. Note that the transfer will not be started until the LX/SX hardware drives the DREQ0# signal active. Write 3H to PCIBAR0 plus A8H (PLX DMA Channel 0 Command/Status register).

2. The LX/SX DMA registers must also be set to transfer 131,072 Lwords.
   - Write 20000H to LX/SX DMA Transfer Size Register at PCIBAR2 plus 8038H. Note the transfer size specifies the number of Lwords transferred.
   - Initialize the SRAM Internal Address register to 0H. External field data is written to SRAM at this starting address. Write 0H to PCIBAR2 plus 8044H.
   - Setting the SRAM DMA channel 0 Threshold register to 1FFFFH will start a DMA demand mode transfer when the internal address counter reaches 1FFFFH (131071 decimal). Write 1FFFFH to PCIBAR2 plus 8048H.
3. Lastly, the LX/SX external rear I/O data is enabled for write to SRAM and DMA Channel 0 demand mode start when the internal address equals the Channel 0 Threshold register.
   - Write SRAM Control Register with 3H to PCIBAR2 plus 8040H.

PCI9656 REGISTERS

PCI9656 Registers (Read/Write) – (PCIBAR0)

The PCI9656 is configured for (PLX PCI9656 chip) C Mode Local bus operation. Only, the PCI9656 User's Manual references to C Mode configuration apply to the LX/SX boards. The LX/SX boards use the (C Mode) generic 32-bit non-multiplexed address and data bus interface for communication between the PCI9656 and the Xilinx FPGA.

PCI9656 USERo CLOCK CONTROL

Note the PCI9656 USERo signal is a general purpose output controlled from the PCI9656 Configuration registers. The "o" at the end of this signal name indicates that this is an output signal.

The default power-up condition of the LX/SX board enables the on-board 66MHz crystal as the active clock. The active clock is used to clock the PCI9656 Local bus, SRAM interface, CPLD operation, and FPGA operation. After configuration, the on board 66MHz can be deselected and a Xilinx FPGA generated clock (PLL_CLK signal) can be selected as the new active board clock. Clock signal selection is controlled via the PCI9656 USERo signal.

The USERo control signal, output from the PCI9656, is used to select between the 66MHz clock and the user defined clock (PLL_CLK). The user defined clock must be defined in the FPGA and output from the FPGA on signal PLL_CLK. The Digital Clock Manager of the FPGA offers a wide range of clock management features including clock multiplication and division for generation of a user defined clock (PLL_CLK). A 66MHz crystal generated clock signal (FPGA_CLK_PLL) is input to the FPGA for use in generation of the user-defined clock signal PLL_CLK. The PLL_CLK can be a minimum of 10MHz and a maximum of 100MHz. Since the PLL_CLK signal is generated and driven by the FPGA, it will only be available after the FPGA is configured. See the example VHDL file included in the engineering design kit and the Xilinx documentation on the Digital Clock Manager for more information.

The USERo signal is controlled via a PCI9656 device register over the PCI bus. The PCI9656 User I/O Control register at PCIBAR0 base address + offset 6CH must have bit-19 set to a logic high to select USERo to be an output from the PCI9656. In addition, User I/O Control register at PCIBAR0 base address + offset 6CH must control bit-16. Bit-16 set to logic high causes the Board clock to be 66MHz. Bit-16 set to logic low will select the PLL_CLK as the Board clock frequency.

Table 3.18: *USERo Control*

PCI Address	PCIBAR0 + 6CH	PCI9656 Runtime Register User I/O Control
Bit-16	Logic "0"	Board clock becomes PLL_CLK
	Logic "1"	Board clock 66MHz (Default)
Bit-19	Logic "0"	Do not set as logic low
	Logic "1"	USERo Enabled as output

Note that the Xilinx FPGA can not be reconfigured with the PLL_CLK signal selected. The first step in FPGA reconfiguration is to clear the FPGA device and this will disable the PLL_CLK signal. Without a board clock the LX/SX board will lock up and require a system power down to reactivate. **The USERo clock control signal must be set to logic high prior to FPGA reconfiguration.**

4.0 THEORY OF OPERATION

This section contains information regarding the design of the board. A description of the basic functionality of the circuitry used on the board is also provided. Refer to the Block Diagram shown in Drawing 4502-054 as you review this material.

PCI INTERFACE LOGIC

A PLX Technology PCI9656 IC installed on the board provides a 66MHz 64-bit interface to the carrier/CPU board per PCI Local Bus Specification 2.2. The interface to the carrier/CPU board allows complete control of all board functions.

The PCI9656 is compliant with both 5V, and 3.3V signaling. The PCI bus VIO signals are tied directly to the PCI9656 chip which monitors the voltage present on VIO to automatically implement the matching signaling voltage.

Note that the LX/SX board requires that system 3.3 volts be present on the PCI bus 3.3V pins. There are some older systems that do not provide 3.3 Volts on the PCI bus 3.3 volt pins. The LX/SX boards will not work in these systems.

This is a master/target board, with the PCI bus interface logic contained within the PCI9656. This logic includes support for PCI commands, including: configuration read/write, and memory read/write. In addition, the PCI interface performs parity error detection, uses a single 2Meg base address register (PCIBAR2), and implements target abort, retry, and disconnect. The logic also implements interrupt requests via interrupt line INTA#.

The PCI9656 becomes the PCI bus master to perform DMA transfers on channels 0 and 1. The DMA control registers of the PCI9656 chip can be configured for DMA block mode and demand mode. The example device driver can be used to exercise DMA block and demand modes of operation. Note that the example program requires the use of continuous burst and slow terminate mode DMA transfer options. For other DMA modes of operation see the PLX Technology PCI9656 user manual.

The DMA demand mode requires Xilinx FPGA hardware to drive two PCI9656 signals active to request the DMA transfer of data. The signal DREQ0# is driven active to request a DMA channel 0 transfer. The signal DREQ1# is driven active to request a DMA channel 1 transfer. To identify the pins corresponding to these signals, see the user constraints (.UCF) file provided in the engineering design kit.

This LX/SX board does not utilize the PCI9656 PCI power management functions. The power management request signal PME# is not used and is tied high with an external pullup resistor.

The internal PCI Arbiter is not used. External pull-up resistors are tied to the REQ[6:1]# input and GNT[6:1]# output signals of the PCI9656 chip.

NOT USED PCI9656 FUNCTIONS

Many features of the PCI9656 are not used in the example design but are available if enabled. It is beyond the scope of this document to duplicate the PCI9656 User's Manual. Please refer to the PCI9656 User's Manual (See Related Publications) for more detailed information.

THEORY OF OPERATION CONTINUED

The PCI9656 is hardwired for "C" bus mode. This is a generic 32-bit non-multiplexed address and data bus interface.

The example design implements the PCI9656 as the local bus master. The local bus is the bus interface between the PCI9656 and the Xilinx FPGA. As the local bus master, the PCI9656 responds to BREQi assertion to relinquish local bus ownership. The example design has BREQi tied low. The Xilinx FPGA does not request the local bus. However, the FPGA may drive BREQi high if the FPGA must take control of the local bus.

The example design implements single cycle mode. In single cycle mode the PCI9656 issues one ADS# per data cycle. The starting address for a single cycle data transfer can be any address.

Burst read and write cycles can be implemented but must be enabled in the PCI9656 and supported in the logic of the FPGA. The PCI9656 Local Address Space 0/Expansion ROM Bus Region Descriptor register at PCIBAR0 base address + offset 18H must have bit-24 set to a logic high to enable bursting.

SYNCHRONOUS Dual-Port SRAM

A 256K x 36-bit synchronous Dual-Port SRAM is provided on the board. One port of the SRAM uses the address and data lines from the PCI9656. The remaining port connects directly to the FPGA. This configuration allows for a continuous data flow from the field inputs through the FPGA to the SRAM and then to the PCI bus via the PCI9656 IC. Both ports of the SRAM operate in Pipeline mode. This allows for faster operational speed but does cause a one-cycle delay during read operations. The pins corresponding to the control signals, address, and data buses are in the user constraint (.UCF) file provided in the engineering design kit.

The SRAM port connected directly to the FPGA supports continuous writes or single cycle reads. The SRAM port connected to the local bus and the PCI9656 IC supports reading and writing using a single cycle or DMA transfers. For single cycle accesses address and control signals are applied to the SRAM during one clock cycle, and either a write will occur on the next cycle or a read in two clock cycles. DMA accesses operate using the continuous burst method for maximum data throughput. The control signal, starting address, and data (if writing) are applied to the SRAM during one clock cycle. Then, during a write DMA transfer, new data is applied by the PCI9656 to the bus every subsequent clock cycle until the transfer is complete. If a DMA read transfer is occurring, then there would be a two-cycle delay before the data was available on the bus. After the initial delay, every subsequent clock cycle will contain the data from the next address.

THEORY OF OPERATION CONTINUED

During DMA transfers the address is held constant on the local bus and incremented internally in the Dual-Port SRAM. Please refer to the IDT70T3519S133BC Data Sheet and PCI9656 Data Sheet (See Related Publications) for more detailed information.

SERIAL EEPROM

A 128 x 16-bit Serial EEPROM is wired to the PCI9656 to provide power-up configuration information required by the PCI9656 device. The stored data in the EEPROM contains PCI device and vendor ID information. In addition, the PCI interrupt line, PCI base address, size and user options such as burst enabled are specified in this memory device. The contents of the serial EEPROM can be changed using the PCI9656 VPD function.

Acromag software also provides the functions needed to implement read and write operations to the serial EEPROM.

CLOCK CONTROL

The board clock is routed to the PCI9656 Local bus pin, SRAM, CPLD, and FPGA using a low skew clock driver (Cypress CY2305). The input to the CY2305 can be one of two sources. The on board 66MHz crystal oscillator is input to the CY2305 upon power-up (as the default condition). After the FPGA is configured, an FPGA generated clock signal (PLL_CLK) can be selected as the board clock. The PLL_CLK signal is selected as the board clock by setting the USERo PCI9656 output signal to logic low. See the "PCI9656 USERo CLOCK CONTROL "section of chapter 3 for additional details.

5.0 SERVICE AND REPAIR

Surface-Mounted Technology (SMT) boards are generally difficult to repair. It is highly recommended that a non-functioning board be returned to Acromag for repair. The board can be easily damaged unless special SMT repair and service tools are used. Further, Acromag has automated test equipment that thoroughly checks the performance of each board. When a board is first produced and when any repair is made, it is tested, placed in a burn-in room at elevated temperature, and retested before shipment.

SERVICE AND REPAIR ASSISTANCE

Please refer to Acromag's Service Policy Bulletin or contact Acromag for complete details on how to obtain parts and repair.

PRELIMINARY SERVICE PROCEDURE

Before beginning repair, be sure that all of the procedures in Section 2, Preparation For Use, have been followed. Also, refer to the documentation of your carrier/CPU board to verify that it is correctly configured. Replacement of the board with one that is known to work correctly is a good technique to isolate a faulty board.

CAUTION: POWER MUST BE TURNED OFF BEFORE REMOVING OR INSERTING BOARDS

WHERE TO GET HELP
www.acromag.com

If you continue to have problems, your next step should be to visit the Acromag worldwide web site at http://www.acromag.com. Our web site contains the most up-to-date product and software information.

Acromag's application engineers can also be contacted directly for technical assistance via telephone or email. Contect information is listed at the bottom of this page. When needed, complete repair services are also available.

6.0 SPECIFICATIONS

Single PMC Board

PHYSICAL

Height	13.5 mm (0.531 in)
Stacking Height	10.0 mm (0.394 in)
Depth	149.0 mm (5.866 in)
Width	74.0 mm (2.913 in)
Board Thickness	1.59 mm (0.062 in)

Unit Weight

PMC-LX/SX: 3.26oz (0.0924Kg), typical

Connectors

- **PMC** PCI Local Bus Interface: Four 64-pin female receptacle header (AMP 120527-1 or equivalent). Three of these connectors interface to the PCI bus the fourth connector provide 64 rear I/O connections.
- Front Field I/O Connector on PMC module (Samtec QSS-075-01-L-D-A) Mating Mezzanine Connector (Samtec QTS-075-01-L-D-A) with 5mm stack height or (Samtec QTS-075-02-L-D-A) with 8mm stack height.

Table 6.1:
Power Requirements for Example Design. Power will vary dependent on the application.

5V Maximum rise time of 100m seconds

Power Requirements		PMC Modules
5V (±5%)	Typical	250mA
	Max.	350mA
3.3V (±5%)	Typical	1200mA
	Max.	1500mA
+/-12V (±5%)	Typical	0mA
	Max.	0mA

On Board 1.2V Power to Virtex-4 FPGA	Current Rating
1.2V (±5%)	7A Maximum

ENVIRONMENTAL

Operating Temperature: 0 to +70°C.
Conduction Cooled PCI mezzanine card: Complies with ANSI/VITA 20-2001 (R2005). The PMC LX/SX, without a faceplate, is fully compatible with a conduction cooled host card.
Relative Humidity: 5-95% Non-Condensing.
Storage Temperature: -55°C to 125°C.
Non-Isolated: Logic and field commons have a direct electrical connection.

Radiated Field Immunity (RFI): Complies with EN61000-4-3 (3V/m, 80 to 1000MHz AM & 900MHz. keyed) and European Norm EN50082-1 with no register upsets.

Conducted R F Immunity (CRFI): Complies with EN61000-4-6 (3V/rms, 150KHz to 80MHz) and European Norm EN50082-1 with no register upsets.

Surge Immunity: Not required for signal I/O per European Norm EN50082-1.

SPECIFICATIONS

Radiated Emissions: Designed to comply with European Norm EN50081-1 for class B equipment. Shielded cable with I/O connections in shielded enclosure are required to meet compliance.

Reliability Prediction

Mean Time Between Failure: MIL-HDBK-217F, Notice 2 @ 25°C.
PMC**-**LX40: 773,246 Hours
PMC**-**LX110: 870,489 Hours
PMC**-**SX35: 869,686 Hours

FPGA (PMC-LX40)

Xilinx XC4VLX40-10FF668C
- 36,864 CLB Flip Flops
- 294,912 Distributed RAM Bits
- 96 18Kbit Block RAMs
- 64 DSP Slices
- 8 Digital Clock Managers

FPGA (PMC-LX60)

Xilinx XC4VLX60-10FF668C
- 53,248 CLB Flip Flops
- 425,984 Distributed RAM Bits
- 160 18Kbit Block RAMs
- 64 DSP Slices
- 8 Digital Clock Managers

FPGA (PMC-SX35)

Xilinx XC4VSX35-10FF668C
- 30,720 CLB Flip Flops
- 245,760 Distributed RAM Bits
- 192 18Kbit Block RAMs
- 192 DSP Slices
- 8 Digital Clock Managers

REAR I/O

The rear I/O P4 PMC connector connects directly to bank 5 of the FPGA. The bank 5 Vcco pins are powered by 2.5 volts and thus will support the 2.5 volt IOStandards. Table 6-38 of the Virtex 4 User Guide lists all the supported IOStandards available. The example design defines the rear I/O with 2.5 volt LVDS.

- Maximum Recommeded Clock Rate…….150MHz (6.7ns clock period)
- Vcco Supply Voltage ……..…..…2.5 volt
- V_{OH} Output High Voltage……..1.602 volt
- V_{OL} Output Low Voltage……..0.898 volt
- V_{ODIFF} Differential Output Voltage ……..…350m volt typical
- V_{OCM} Output Common Mode Voltage…...1.25 volt typical
- V_{IDIFF} Differential Input Voltage……..100m volt minimum
- V_{ICM} Input Common Mode Voltage……0.3 volt min, 2.2 volt max

FRONT I/O

See the mezzanine module users manual for front I/O specifications. This PMC module uses the 150 pin Samtec connector part number XX which mates with the mezzanine module connector part number QTS-075-02-L-D-A-K

Write Disable Jumper

Write Disable Jumper: Removal of surface mount resistor R171 disables write to the PCI bus configuration EEPROM and removal of surface mount resister R172 disables write to the Xilinx FPGA configuration flash device.

Board Crystal Oscillator: 66MHz
Frequency Stability: ± 0.0025% or 25ppm

Double Data Rate SDRAM

32M x 36-bit Density
Micron MT46V32M16BN-5B

SDRAM Crystal Oscillator: 200MHz
Frequency Stability: ± 0.01% or 100ppm

DDR SDRAM Clock: 150MHz
Data Transfer Rate: 300M longwords(32-bits)/s

Dual Port SRAM

256K x 36-bit Integrated Devices Technology IDT70T3519S133BC, 133 Megahertz Speed

PLX 9656 Local Bus

LCLK Maximum Clock Frequency: 66MHz

PCI Local Bus Interface

PMC Compatibility: Conforms to PCI Bus Specification, Revision 2.2 and PMC Specification, P1386.1

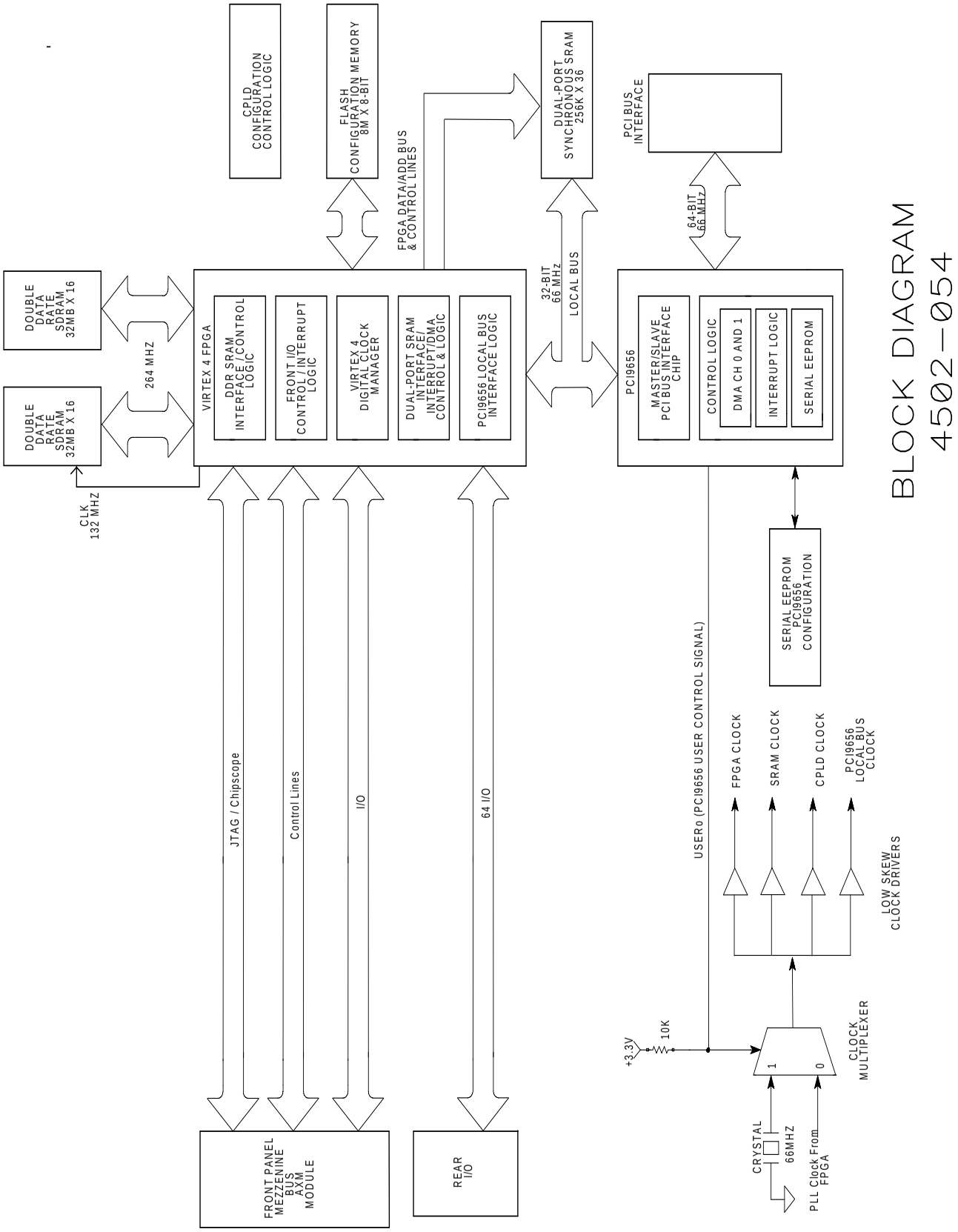
PCI Master/Target: Implemented by PLX Technology PCI9656 Chip

2M Memory Space Required: One Base Address Register for this 2M space.

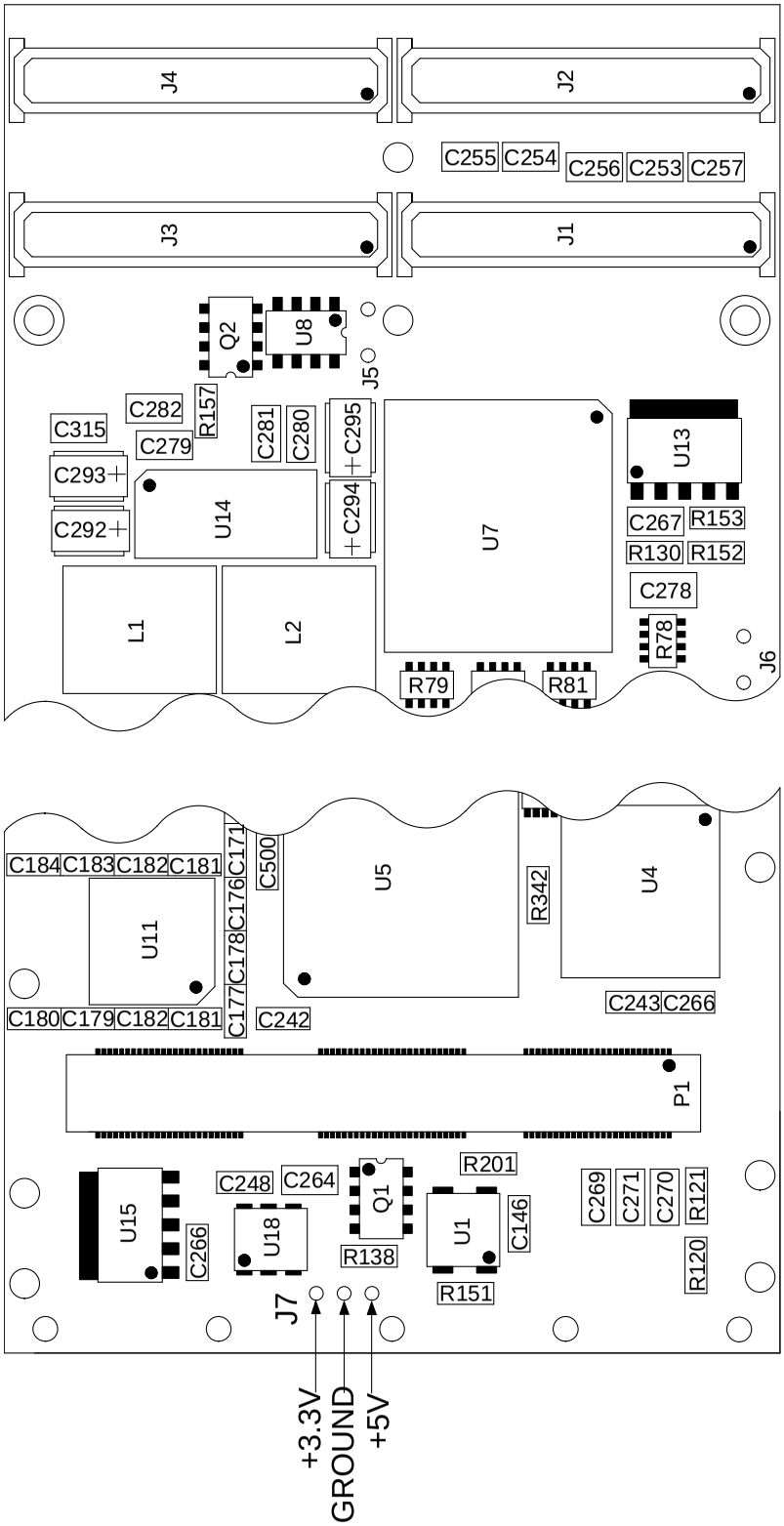
PCI commands Supported: Configuration Read/Write memory Read/Write: 64,32,16, and 8-bit data transfer types supported.

Signaling: 5V or 3.3V Compliant

INTA#: Interrupt A is used to request an interrupt. Source of interrupt can be from the Digital I/O, or PCI9656 Functions.



BLOCK DIAGRAM
4502-054



FRONT PANEL

J7 EXTERNAL POWER LOCATION
4502-053

In standalone mode, where the card is not plugged into the PCI bus, external power must be provided via the J7 contact holes.